

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

Claim 1 (canceled)

Claim 2 (previously presented): The method of claim 8 wherein the first log reader launches the second log reader.

Claim 3 (previously presented): The method of claim 8 further comprising: terminating the second log reader.

Claim 4 (previously presented): The method of claim 8 wherein the first log reader and the second log reader execute separately.

Claim 5 (canceled)

Claim 6 (canceled)

Claim 7 (canceled)

Claim 8 (previously presented): A method comprising:  
publishing, by a first log reader, first messages to a plurality of queues, the first messages comprising changes for transactions extracted from a log by the first log reader;  
in response to one of the queues becoming unavailable, launching a second log reader to read and extract from the log, and to publish second messages comprising changes for transactions extracted from the log by the second log reader to the unavailable queue as a catch-up queue; and

in response to the catch-up queue becoming available and the second log reader reaches the end of the log, transferring the publishing of the second messages for the catch-up queue from the second log reader to the first log reader,

wherein said transferring comprises re-synchronizing the second log reader and the first log reader,

wherein said re-synchronizing comprises:

sending synchronization messages between the first log reader and the second log reader; a first synchronization message of the synchronization messages being sent from the second log reader to the first log reader with second log reader restart information; and

in response to the second log reader being behind the first log reader in reading the log based on the second log reader restart information, a second synchronization message of the synchronization messages being sent from the first log reader to the second log reader, the second synchronization message comprising first log reader restart information having a first-last-queue-commit point timestamp, and suspending the publishing by the first log reader until the second log reader reaches the first-last-queue-commit point timestamp or the end of the log.

Claim 9 (previously presented): A method comprising:

publishing, by a first log reader, first messages to a plurality of queues, the first messages comprising changes for transactions extracted from a log by the first log reader;

in response to one of the queues becoming unavailable, launching a second log reader to read and extract from the log, and to publish second messages comprising changes for transactions extracted from the log by the second log reader to the unavailable queue as a catch-up queue; and

in response to the catch-up queue being available and the second log reader reaches the end of the log, transferring the publishing of the second messages for the catch-up queue from the second log reader to the first log reader,

wherein said transferring comprises re-synchronizing the second log reader and the first log reader,

wherein said re-synchronizing comprises:

sending synchronization messages between the first log reader and the second log reader;

a first synchronization message of the synchronization messages being sent from the second log reader to the first log reader with second log reader restart information; and

in response to the second log reader being ahead of the first log reader in reading the log based on the second log reader restart information, a stop message of the synchronization messages being sent to the second log reader;

in response to the second log reader being behind the first log reader in reading the log based on the second log reader restart information, a second synchronization message of the synchronization messages being sent from the first log reader to the second log reader, the second synchronization message comprising first log reader restart information having a first-last-queue-commit point timestamp, and suspending the publishing by the first log reader until the second log reader reaches the first-last-queue-commit point timestamp or the end of the log.

Claim 10 (currently amended): The method of claim 8 further comprising:

storing the first log reader restart information, associated with the first log reader, in persistent memory;

storing the second log reader restart information, associated with the second log reader, in the persistent memory;

stopping the first log reader and the second log reader,

first launching the first log reader based on the first log reader restart information; and

second launching the second log reader based on the second log reader restart information.

Claims 11-30 (canceled)

Claim 31 (original): The method of claim 9 wherein the first log reader launches the second log reader.

Claim 32 (original): The method of claim 9 further comprising:

terminating the second log reader.

Claim 33 (original): The method of claim 9 wherein the first log reader and the second log reader execute separately.

Claim 34 (currently amended): The method of claim 9 further comprising:

- storing the first log reader restart information, associated with the first log reader, in persistent memory;
- storing the second log reader restart information, associated with the second log reader, in the persistent memory;
- stopping the first log reader and the second log reader,
- first launching the first log reader based on the first log reader restart information; and
- second launching the second log reader based on the second log reader restart information.